

REMARKS

The application was filed on 03 April 2000 with sixteen claims. In the first Examiner's Office Action mailed 14 February 2003, the Examiner rejected claim 12 under 35 U.S.C. §102(b) as being anticipated by U.S. Patent 6,249,291 B1 entitled "Method and Apparatus for Managing Internet Transactions" to Popp et al. (Popp '291). The Examiner further rejected claims 1-11 and 13-16 under 35 U.S.C. §103(a) as being unpatentable over U.S. Patent 6,368,273 B1 entitled "Networked System for Interactive Communication and Remote Monitoring of Individuals" to Brown (Brown '273) in view of U.S. Patent 6,516,353 B1 entitled "System and Method for Interactive EDI Transactions" to Richards (Richards '353). In response, Applicant amended claims 1, 4, 5, 6, 13, and 16, and cancelled claim 2 to represent that the script remains and executes on the server. Applicant amended claims 1 and 12 to represent that different policies of different desktop/containers are contemplated within the claim and thus those user-interface components are downloaded within the unique policy framework of each desktop/container.

The Examiner issued a final rejection of claims 1-11, 13-16 under 35 U.S.C. §103(a) as being unpatentable over Brown '273 in view of Richards '353. Claims 12 was finally rejected under 35 U.S.C. §103(a) as being unpatentable over Popp '291 in view of Brown '273. In response, Applicants have filed a Request for Continued Examination under 35 U.S.C. §1.114 with an amendment. Claims 1, 3-16 are pending.

The Rejection of claims 1-11, 13-16 under 35 U.S.C. §103(a)

The Examiner further rejected claims 1-11 and 13-16 under 35 U.S.C. §103(a) as being unpatentable over Brown '273 in view of Richards '353. In response, Applicant notes that claim 2 was previously cancelled and now amend claims 1, 4-6, 12, 13, 16 to particularly point out and distinctly claim that the script executes on the server and that only those user interface components associated with a particular task are downloaded to the client within its policy/framework. Upon completion of a task association with the application, the user interface components are closed before the user-interface components associated with the next task are downloaded to the container/desktop of the client. Thus, a scripted task comprises more than one

user interface component and these interface components are downloaded piecemeal, i.e., when needed, but the script still executes on the server. The claimed invention further contemplates that different clients will have different container/desktops each with its policy framework; within this menagerie of clients, the script selects and downloads only those user-interface components within the policy/framework of the individual client.

The Examiner references Brown '273 as having the elements of the claimed invention, except for the closing of the user interface components on a task-basis, which teaching the Examiner alleges is provided by Richards '353. Respectfully, Brown '273 discloses a medical monitoring software application residing on a server which application can be called from a client. The server downloads the entire script to the client apparatus and the script executes on the apparatus to inquire as to the state of a medical patient. Again and again, Brown '273 states that the application and the script are downloaded to the client apparatus and the client apparatus executes the application and the script. *See* Brown '273 at column 4, lines 52-54 (Each apparatus is designed to interact with a patient in accordance with **script programs received from the server.**); column 5, lines 14-16 (The **script programs are executed by each apparatus** to communicate queries and messages to a patient, ....); column 5, lines 23-25 (Each remote **apparatus is designed to execute assigned script programs which it receives from server.**); column 5, lines 65-67 (The firmware includes a script interpreter used by **microprocessor [within the apparatus] to execute the script programs.**); column 6, lines 22-25 (The data [received by the apparatus through a modem] includes **script programs which are received from the server** ....); column 9, lines 53-55 (In step 218, **server transmits the assigned script program to the patient's apparatus** through communication network.) and again at column 14, lines 11-14 (Next, server retrieves the assigned script program from database and **transmits the script program to the individual's apparatus** through communication network. ¶ The **apparatus receives and executes the script program.**) [reference numbers omitted and emphasis added]. Thus, Brown '273 simply downloads a script to an apparatus.

Applicant explains the disadvantages of downloading an entire script to the client in the originally filed specification which states that Applicant's "invention solves the problem of downloading large amounts of executable code to implement the user-interface of an application

in several ways. First and most important, the script executes on the server and is not downloaded to the client.” Attorney for the Applicant cannot make it more clear that this distinction between Brown ‘273 and the claimed invention is significant and is not obvious, in view of Brown ‘273, by itself or in combination with Richards ‘353 and/or Popp 291.

Brown ‘272, moreover, discloses only one apparatus as shown in Figure 3; this apparatus has only one policy/framework. Thus, another distinction between Brown ‘273 and the claimed invention is that Brown ‘273 does not disclose a **software application having a plurality of policy frameworks, each associated with a respective one of the plurality of clients**. The invention, as claimed, contemplates that the user interface components will be different for clients having different policies/frameworks, i.e., a client may be a personal or larger computer system; a notebook or laptop computer; a network terminal or a thin client or other terminal-like devices having smaller memory; voice response units; terminals, world wide web browsers; and even pervasive mobile devices, such as personal digital assistants, pagers, and cell phones. Brown ‘273 has only one apparatus and does not disclose different container/desktops.

The Examiner continues and admits that Brown ‘273 does not explicitly disclose the script executing and closing the user-interface components; and the script executing and closing the another user-interface component. The Examiner, however, asserts that the use and advantages for executing such scripts are well known as evidenced by the teachings of Richards ‘353 at column 5, lines 30-36, 40-42. Respectfully, Richards ‘353 at column 5, lines 30-42, teach the process of creating or building and editing a script, not how the script opens and closes sequential user-interface components. Quite simply, Applicant poses the question: If, in Brown ‘273, the entire script is downloaded to and executes on the container/desktop, why would user-interface components be opened and closed to save memory? The memory of the container/desktop of Brown ‘273 already contains all the user-interface components - there is no need to download piecemeal the user-interface components of each task, then open and then close them when the task is completed.

Respectfully, Applicant requests the Examiner to withdraw the rejection of claims 1, 3-11, 13-16 under 35 U.S.C. §103(a) using the alleged combination of Brown ‘273 with Richards ‘353. The Rejection of claim 12 under 35 U.S.C. §103(a)

The Examiner further rejected claim 12 under 35 U.S.C. §103(a) as being unpatentable over Popp ‘291 in view of Brown ‘273. The Examiner asserts that Popp ‘291 discloses a method to script user-interface components together to create an application stored on a server and whose user-interface components are downloaded to one of a variety of container/desktops of different clients. The Examiner then states that Brown ‘273 discloses a script program wherein the server uses a patient’s unique identification code which corresponds to the type of device the patient uses to identify a specific script program to service the patient. Thus, the Examiner reasons, that one of ordinary skill the art would be compelled to combine Popp ‘291 with Brown ‘273 to implement or incorporate a policy framework being unique to one of said container/desktops of said client and in accordance with the policy framework unique to the container/desktop of said client in Popp’s ‘291 method in order for each remote apparatus or device to execute assigned script programs which it receives from the server. Applicant asserts that neither reference discloses the claimed language of **each one policy framework being unique to one of said variety of container/desktops of different clients**, as in claim 12. Neither reference, moreover, suggests the alleged combination proffered by the Examiner. First, Brown ‘273 discloses only one apparatus, thus Brown ‘273 has only one client shown in its Figure 3 and only one policy framework. Contrary to the Examiner’s analogy, the policy/framework disclosed by Applicant is NOT the same thing as different patients’ profiles, as disclosed by Brown ‘273. According to Applicant, *see* page 16, lines 12 through page 17, lines 12, the policy/framework of a desktop/container takes into account whether the container/desktop terminates each task when the transition to the next task occur or when the parent task terminates; whether a single task, a fixed number, or a user controlled variable number of tasks can be executed simultaneously; the number of user-interface components that can be viewed concurrently; the display policy regarding position, resizing, cropping user-interface components; whether single display panel, multiple tiled panels, or overlapping windows will be implemented, etc. Quite frankly, the application remains the same. The application of Brown ‘273 must change depending upon the patient profile, i.e., does the patient have diabetes or asthma? The application and the script must change depending upon the health of the patient; not on the policy framework of the client. Thus, neither

Popp '291 nor Brown '273 discloses creating a script of an application based on the policy framework, i.e., the software/hardware, of the client without affecting the application code.

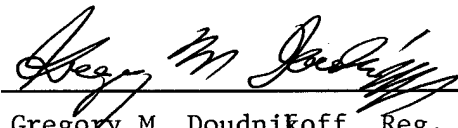
Conclusion

In view of the final rejection of claims 1, 3-16, Applicants have amended the claims and have filed a Request for Continued Examination. Thus, now having further distinguished Applicant's claims from the references of Brown '273, Richards '353, and Popp '291 by clearly claiming that the script executes on the server, and that the script downloads to a container/desktop only those user-interface components within the policy framework of a particular client, Applicant respectfully request the Examiner to review the amendments and the remarks and to pass the application to issuance.

Respectfully submitted,

J. Julian Paas

Date: 03 November 2003

By   
Gregory M. Doudnikoff Reg. No., 32,847